-78-61

ABS ONLY

167042

2P.

N89-16297

Implementation of Ada Protocols on MIL-STD-1553 B Data Bus

by
Smil Ruhman and Flavia Rosemberg
Department of Applied Mathematics
The Weizmann Institute of Science
Rehovot, Israel

Standardization activity of data communication in avionic systems started in 1968 for the purpose of total system integration and the elimination of heavy wire bundles carrying signals between various sub-assemblies. First issued in 1973, MIL-STD-1553 (USAF) replaced point-to-point wiring with a digital time-multiplexed common-bus for serial data transmission. Reissued in 1975 as a tri-service standard (version A) and again revised in 1978 (version B), it came into wide use and is supported by integrated hardware. However a major development effort must still be invested in every real-time system for interprocessor synchronization and scheduling of information transfer in the absence of a high-level language possessing communication constructs.

The growing complexity of avionic systems is straining the capabilities of MIL-STD-1553 B, but a much greater challenge to it is posed by Ada, the standard language adopted by the US Department of Defense for real-time, computer-embedded-systems. The stochastic, distributed nature of Ada with its rendez vous protocol for interprocess synchronization is not matched well by the deterministic central control of communication in MIL-STD-1553 B. Accordingly, the authors have proposed hardware implementation of Ada communication protocols in a contention/token bus or token ring network {1}.

However, during the transition period when the current command/response multiplex data bus is still flourishing and the development environment for distributed multi-computer Ada systems is as yet lacking, a temporary accommodation of the standard language with the standard bus could be very useful and even highly desirable. By concentrating all status information and decisions at the Bus Controller, it was found possible to construct an elegant and efficient hardware implementation of the Ada protocols at the bus interface, and this solution is the subject of our paper. No compromises are taken with the bus standard, and no changes imposed on Remote Terminals. Implementation hardware is restricted to the Bus Controller and its alternate, the Bus Monitor.

The idea is based on polling of the Remote Terminals by the Controller for entry calls, accept statements, or results (output parameters). The Controller interface maintains all the entry call queues and the list of ready accept statements, searches for a match, and issues the appropriate commands for transfer or execution depending on the presence of input and/or output parameters. In addition, the Controller interface times the delays of selective waits and of timed entry calls, and controls the execution of delay alternatives and of "else" clauses. Most of these operations are clearly of a match-making or associative nature. To avoid long Controller response

ORIGINAL PAGE IS
OF POOR QUALITY

times due to conventional searching of extensive files, all queues and lists are stored in a common associative memory which is microsequenced from a control store. The paper presents the algorithms employed, defines the command and data formats, and outlines the hardware organization. The resulting bus traffic and speed of operation are discussed. It is interesting to note that while our algorithms take advantage of mode commands to reduce traffic, no such use was found for broadcast commands.

The proposed approach renders distributed intertask synchronization transparent to the designer and implements it in hardware at the bus interface. In addition, data buffering becomes unnecessary, since transfer is delayed until both parties are ready. Many important advantages result, chief among them being: facilitation of the development environment; major savings in specific development effort; conservation of system resources such as host processing and line transmission capacity; and faster system response.

Reference:

{1} Rosemberg, F. and S. Ruhman, "Hierarchical partitions in cyclic closed systems: a hardware oriented approach", Proceedings of Computers in Aerospace V Conference, Longbeach, CA, October 1985, pp. 148-155.